



# GCSE Examiners' Report

Computer Science  
GCSE  
Summer 2024

## Introduction

Our Principal Examiners' report provides valuable feedback on the recent assessment series. It has been written by our Principal Examiners and Principal Moderators after the completion of marking and moderation, and details how candidates have performed in each unit.

This report opens with a summary of candidates' performance, including the assessment objectives/skills/topics/themes being tested, and highlights the characteristics of successful performance and where performance could be improved. It then looks in detail at each unit, pinpointing aspects that proved challenging to some candidates and suggesting some reasons as to why that might be.<sup>1</sup>

The information found in this report provides valuable insight for practitioners to support their teaching and learning activity. We would also encourage practitioners to share this document – in its entirety or in part – with their learners to help with exam preparation, to understand how to avoid pitfalls and to add to their revision toolbox.

## Further support

Document	Description	Link
Professional Learning / CPD	WJEC offers an extensive programme of online and face-to-face Professional Learning events. Access interactive feedback, review example candidate responses, gain practical ideas for the classroom and put questions to our dedicated team by registering for one of our events here.	<a href="https://www.wjec.co.uk/home/professional-learning/">https://www.wjec.co.uk/home/professional-learning/</a>
Past papers	Access the bank of past papers for this qualification, including the most recent assessments. Please note that we do not make past papers available on the public website until 12 months after the examination.	<a href="#">Portal by WJEC</a> or on the WJEC subject page
Grade boundary information	Grade boundaries are the minimum number of marks needed to achieve each grade. For unitised specifications grade boundaries are expressed on a Uniform Mark Scale (UMS). UMS grade boundaries remain the same every year as the range of UMS mark percentages allocated to a particular grade does not change. UMS grade boundaries are published at overall subject and unit level.  For linear specifications, a single grade is awarded for the subject, rather than for each unit that contributes towards the overall grade. Grade boundaries are published on results day.	For unitised specifications click here: <a href="#">Results, Grade Boundaries and PRS (wjec.co.uk)</a>

---

<sup>1</sup> Please note that where overall performance on a question/question part was considered good, with no particular areas to highlight, these questions have not been included in the report.

Exam Results Analysis	WJEC provides information to examination centres via the WJEC Portal. This is restricted to centre staff only. Access is granted to centre staff by the Examinations Officer at the centre.	<a href="#">Portal by WJEC</a>
Classroom Resources	Access our extensive range of FREE classroom resources, including blended learning materials, exam walk-throughs and knowledge organisers to support teaching and learning.	<a href="https://resources.wjec.co.uk/">https://resources.wjec.co.uk/</a>
Bank of Professional Learning materials	Access our bank of Professional Learning materials from previous events from our secure website and additional pre-recorded materials available in the public domain.	<a href="#">Portal by WJEC</a> or on the WJEC subject page.
Become an examiner with WJEC.	We are always looking to recruit new examiners or moderators. These opportunities can provide you with valuable insight into the assessment process, enhance your skill set, increase your understanding of your subject and inform your teaching.	<a href="#">Become an Examiner   WJEC</a>

## Contents

	<b>Page</b>
Executive summary	5
Unit 1 – Understanding Computer Science	7
Unit 2 – Computational Thinking and Programming	9
Unit 3 (NEA) – Software Development	10
Supporting you – useful contacts and links	13

## Executive Summary

The performance of candidates in the GCSE Computer Science Summer 2024 assessment series was generally positive, showing improvement compared to previous years.

In Unit 1 - Understanding Computer Science, which contributes the majority of the qualification, candidates performed well overall on questions involving number representations, Boolean algebra, and application of concepts like encryption and programming languages. Many were able to score high marks on these topics, displaying a solid grasp of the skills.

Candidates, did however, find some areas more challenging. Describing the purpose and functionality of the TCP/IP model proved difficult, with only a minority providing detailed, accurate explanations. Questions requiring descriptions of IDE tools, utility software, and legislation also had lower marks, suggesting candidates struggled in addressing the specific requirements.

In Unit 2 - Computational Thinking and Programming, candidates generally showed a good understanding of the specification, with many excellent answers and much improved responses to the Java (Greenfoot) programming questions. They also demonstrated skills in areas such as HTML tags, file naming, and object-oriented programming concepts.

Some candidates, however, provided incorrect solutions, one such example was using assembly code instead of pseudocode when asked for within algorithm question. Implementing the specific requirements of the Java (Greenfoot) programming tasks also proved challenging for some candidates.

For the NEA Unit 3 - Software Development, good examples of programmed solutions were seen, but not all candidates could fully meet the scenario requirements. Creating user interfaces was a relative strength, but validation routines often lacked clear instructions. Many struggled with effective problem analysis, justifying language choices, describing all processing stages before coding, and developing focused testing strategies targeting objectives and logic. Providing valid, detailed suggestions for further improvements was also an area for development.

Overall, the examination series identified areas of strength like number representations and much improved Java (Greenfoot) programming. Areas for improvement are areas such as focused technical descriptions, algorithmic thinking, and refining/evaluating programmed solutions.

Areas for improvement	Classroom resources	Brief description of resource
General	<a href="#">Portal (WJEC)</a>	Past papers, marking schemes and exemplar materials
General – administration	<a href="#">WJEC Website</a>	Specification
General – administration	<a href="#">WJEC Website</a>	Teacher guidance
Unit 1	<a href="#">Digital Resources</a>	Blended Learning

Unit 1	Digital resources <a href="#">(Editable)</a>   <a href="#">(PDF)</a>	Theory booklet
Unit 2 - Greenfoot	<a href="#">Digital resources – tutorial videos</a>	Greenfoot tutorials
Units 1 & 2	<a href="#">Digital resources</a>	Knowledge organisers
Units 1 & 2	<a href="#">Ada Computer Science</a>	Theory website

# COMPUTER SCIENCE

## GCSE

Summer 2024

### UNIT 1 – UNDERSTANDING COMPUTER SCIENCE

#### Overview of the Unit

The unit 1 assessment covers AO1 'Demonstration of knowledge and understanding of key concepts and principles' and AO2 'Application of knowledge and understanding of key concepts and principles'. In order to comply with the qualification specification, and be consistent with previous versions of the assessment, 52 of the marks were allocated to AO1 and 48 of the marks to AO2.

The content of the assessment in relation to AO1 covered most of the main topics in the specification, including CPU architecture, networks, program compilation, software engineering (IDE Tools), utility software, legislation and high level/ low level programming languages.

The content of the assessment for AO2 included the identification of program errors, number conversions, binary representation of sound and application of Boolean logic. There were also part questions on setting of data types, validation rules, encryption, and use of high level/ low level programming languages.

The performance across the assessment was broadly in line with previous series, with a mean mark of 37 out of 100, compared with a mean mark of 36 in 2023 and of 42 in 2019. For the AOs the facility factors for individual questions indicate that the areas requiring descriptions, or an explanation proved to be more challenging than the application tasks required for AO2.

#### Comments on individual questions/sections

The initial tick box type questions on CPU components and network hardware provided the intended positive start to the paper, with facility factors over 0.5 and attempted rates of virtually 100%.

The AO1 questions requiring descriptions of IDE tools, utility software and legislation proved to be more challenging, with facility factors in the region of 0.3 – 0.4 and attempted rates of less than 85%. When addressing these topics, the more successful candidates noted the emphasis of the questions and drafted responses that concentrated on the specific requirements, such as tools to help identify and remove errors in program code. Less successful candidates tended to overlook the specific requirements of the questions and produce responses that covered the topics in more general terms, with, for example, references to text editing and program translation when describing IDE tools.

The band marked AO1 question on the purpose and functionality of the TCP/IP 5-layer model produced a facility factor in the region of 0.2 and an attempted rate of 75%. This complex and technical topic produced responses ranging from detailed and accurate descriptions of functionality to simple lists naming some of the layers. Candidates who took the time to plan their work, noting down key points before starting any detailed response, tended to be the more successful, although few explanations that covered the purpose of the model were seen.

For AO2 written responses, question 9, part (c) was the only other 'explain' question on the paper. The question concerned data encryption and also produced an attempted rate of 75%. Question 10(b), which required descriptions of situations where high level or low-level languages would be used produced a facility factor of 0.1, which is the lowest for the paper.

More accessible questions tended to involve number representations or application of logic, with question 4 on hexadecimal numbers producing a facility factor of 0.7, question 8 (c) on Boolean algebra with a facility factor of 0.6, and questions 8 (d) truth table and 11 (a) binary representation both with facility factors of 0.8. The attempted rate for each of these questions being over 95%. In general, the candidates had been well prepared to address these topics, and most were able to score high marks in these areas.

# COMPUTER SCIENCE

## GCSE

Summer 2024

### UNIT 2 – COMPUTATIONAL THINKING AND PROGRAMMING

#### Overview of the Unit

Most candidates demonstrated a good understanding of the specification. Many candidates were well prepared, and many excellent answers were evident. There was evidence also that most candidates had been well prepared for some of the practical programming elements.

It was pleasing to see the Java programming questions answered so well. With programmed screen savers and interactive games in evidence.

#### Comments on individual questions/sections

- Q.1** We would encourage candidates to test the HTML tags for themselves even if they are sure what the tags do. This may help them to ensure what they present is functional and provides the formatting required.
- Q.2** Some candidates did not complete the final file name. However, this was largely answered well by most candidates. However, a significant number of candidates found the link (using `<a href=. . .`) difficult. Candidates also found the need to use `<ul>` and `<li>` tag difficult. As standard, candidates also forgot to close the HTML tags resulting in incorrect formatting. Candidates should not use generated code or CSS as the specification states the HTML tags that should be used. Some candidates confused the Head, Title and Body tags. Use of `HTTP(s)://` was again limited across centres.
- Q.3** Was largely well attempted by candidates and showed notable improvement.
- Q.4** c - Lots used assembly code, if they are asked to solve an algorithm it needs to be in Pseudo code /Python. Also, candidates could be encouraged to look for and ensure they include:  
all possible inputs user requests and to variables, use of loops, include a count +=  
and include all possible outputs from variables.
- Q.5, 6 & 7** Java programming tasks. Some candidates do not follow the guidance given in the paper for the game or the screensaver and implement different game decisions, however it was pleasing to see a good understanding of object-oriented programming at GCSE level.

## UNIT 3 (NEA) – SOFTWARE DEVELOPMENT

### Overview of the Unit

This unit requires learners to produce a programmed solution to a problem. They must analyse the problem, design a solution to the problem, develop a final programmed solution, test the solution and give suggestions for further development of the solution. Throughout the production of the solution learners are required to produce a refinement log that evidences the development of the solution.

Examples of good work were seen during moderation this summer. It should be noted that it is essential that candidates undertake the correct scenario for the current series.

This unit represents 20% of the qualification and comprises of the below AOs:

- AO2 - 2%  
Apply knowledge and understanding of key concepts and principles of computer science.
- AO3 - 18%  
Analyse problems in computational terms to make reasoned judgements and to design, program, evaluate and refine solutions.

### Tasks

#### Comments on tasks/questions relating to candidate performance/meeting assessment criteria

The scenario has a bullet pointed lists of requirements. To access full marks for the implementation of the solution to the given problem, all bullet points should be covered. However, many candidates were not able to produce a solution that covered all bullet points of the scenario.

The scenario was based on the creation of an order and invoice system for Park Vale Chocolates. The brief required candidates to create software that allowed for ordering of tins of chocolates in specific quantities as well as other requirements, i.e.:

- The contents of the tins must add up to 1 kg.
- Customers can choose up to 500g of any one flavour.
- They could add flavours in increments of 100g.
- The system would have fixed costs of £9.99 per tin and a delivery fee of £4.99.
- An additional charge of 10p per letter for a printed message inside the tin.

The system would be required to record customer orders and create a dispatch note for the packing staff which would either include a personalised note or not for inclusion within the tin.

Most candidates could create an interface that allowed users to enter the customer's personal details when purchasing the tin. Many candidates were able to produce code that could carry out the required functions. Candidates also created validation routines. However, these routines often caused issues when running the code as the interfaces did not provide clear enough instructions for the end user.

## **Task marking**

### **Comments on approaches to internal marking**

Most centres made good use of the centre comments section of the candidate declaration forms, however, this could be further improved with more detailed comments. Several centres include the band marks on a per candidate basis that have been highlighted to demonstrate where the candidate has obtained the marks. Few centres add annotation to the candidates work.

The accuracy of marking has improved year on year, and we would welcome centres to attend the live CPD sessions or webinars in the Autumn.

Requirements for the Report:

The specification states that the candidates should produce a report that:

- analyses the given information
- includes a design of a solution to the given problem
- programming of a solution to the given problem
- testing and refinement of the application, noting the refinements in the refinement log
- gives an evaluation of the application

### **Refinement log**

Candidates are required to complete a log of their activities during the twenty-hour controlled assignment. Almost all candidates presented a completed log. However, a significant number of candidates submitted logs that included many copied and pasted entries. Where candidates had made effective use of their logs; entries included discussion of problems encountered and solutions to these problems. Many candidates were able to identify action points for following sessions that would enable them to make more effective use of their time.

### **Scope of the problem**

A minority of candidates presented effective analysis of the given scenario while many either restated the problem or copied and pasted the contents of the brief. Most candidates were able to outline the objectives for their solution to the given problem.

### **Design of Solution**

In a significant minority of instances, candidates were neither able to justify their choice of programming language using appropriate technical terminology nor relate the features and facilities of the language to their proposed method of solution. Many candidates were able to describe some of the process stages required for their solutions in pseudo code and/or flowcharts. However, fewer candidates covered all processing stages for their proposed solution. In several cases it was not clear that this work had been completed before implementation. Retrospective designs will not be given any credit at moderation.

### **Effectiveness of solution**

Most candidates had produced a solution that allowed the user to enter the details of the businesses and customers into the system. A large percentage of candidates developed solutions using the Python programming language, while others produced solutions using a visual programming language. The quality of the interfaces produced varied considerably. Centres are reminded that the use of online software development tools is prohibited and that 'drag and drop' block type IDEs is not adequate. Most solutions were modular and included the required authentication routines.

### **Technical quality**

Many candidates produced code that was self-documenting and there were instances of the code being well structured. Where candidates had a good understanding of the language they were using, there was evidence of the use of consistent style throughout including indentation and use of white space. In general candidates produced code that used meaningful identifiers and appropriate constants and had coded some validation routines. Many candidates included some annotation of their code with more able candidates included annotation that demonstrated their understanding of the problem and solution.

### **Test strategy**

Most candidates were able to describe some type of testing strategy and some evaluation criteria. In future candidates may benefit from considering their objectives when describing their testing strategies and evaluation criteria, ensuring that they plan to test and evaluate against each objective.

### **Testing**

Most candidates were able to design tests that would demonstrate the functioning of parts of their solution. However, some test plans would have benefited from focusing on the logic of the solution rather than repeatedly testing the less complex parts of the system.

In future candidates would benefit from using their objectives and success criteria as a framework for their test plans and ensure that these are met by their solution.

As solutions should be able to carry out particular functions, candidates should ensure that the data entered produces the correct result and the output is correctly formatted. The scenarios provided many opportunities for candidates to test the logic of their solutions using text-based data to produce a mathematical result to their calculations.

All tests from the test strategy should be evidenced with screenshots and discussed rather than stating if a particular test was a pass or fail.

### **Further development**

The specification calls for candidates to:

- consider the outcomes of the testing process in terms of how well the application meet the objectives set at the beginning of the project
- describe the good features of the application and identifies areas for further development
- provide detailed suggestions for specific extensions to the application.

Many of the discussions produced by the candidates were brief and tended to be narrative rather than reflective and evaluative in nature.

Few candidates offered valid and detailed suggestions for future improvements. However, a minority were able to discuss their solutions in the light of their structure and suggest viable improvements that could be created using their chosen language.

## Supporting you

### Useful contacts and links

Our friendly subject team is on hand to support you between 8.30am and 5.00pm, Monday to Friday.

Tel: 02920 265 401

Email: [CS@wjec.co.uk](mailto:CS@wjec.co.uk)

Qualification webpage: [GCSE Computer Science \(wjec.co.uk\)](https://www.wjec.co.uk/gcse-computer-science)

See other useful contacts here: [Useful Contacts | WJEC](#)

### CPD Training / Professional Learning

Access our popular, free online CPD/PL courses to receive exam feedback and put questions to our subject team, and attend one of our face-to-face events, focused on enhancing teaching and learning, providing practical classroom ideas and developing understanding of marking and assessment.

Please find details for all our courses here: <https://www.wjec.co.uk/home/professional-learning/>

### WJEC Qualifications

As Wales' largest awarding body, WJEC supports its education community by providing trusted bilingual qualifications, specialist support, and reliable assessment to schools and colleges across the country. This allows our learners to reach their full potential.

With more than 70 years' experience, we are also amongst the leading providers in both England and Northern Ireland.



WJEC  
245 Western Avenue  
Cardiff CF5 2YX  
Tel No 029 2026 5000  
Fax 029 2057 5994  
E-mail: [exams@wjec.co.uk](mailto:exams@wjec.co.uk)  
website: [www.wjec.co.uk](http://www.wjec.co.uk)